Machine Learning 1

Lecture 10.3 - Unsupervised Learning
Probabilistic Principal Component Analysis

Erik Bekkers

(Bishop 12.2.1)

*Slide credits: Rianne van den Berg*

Probabilistic PCA

‣ Probabilistic view of PCA:

   ‣ Learn it via maximum likelihood

   ‣ (Third) alternative view of PCA

   ‣ Both latent and observed variables are Gaussian

Continuous latent variable model

‣ Data: $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}, \boldsymbol{x}_n \in \mathbb{R}^D$

‣ Goal: learn a $M < D$ **continuous latent space** by maximizing the likelihood of the probabilistic model, M given

‣ Recall the continuous latent variable model

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z} = \int p(\boldsymbol{x}|\boldsymbol{z}) p(\boldsymbol{z}) d\boldsymbol{z}$$

(Bishop 12.9)
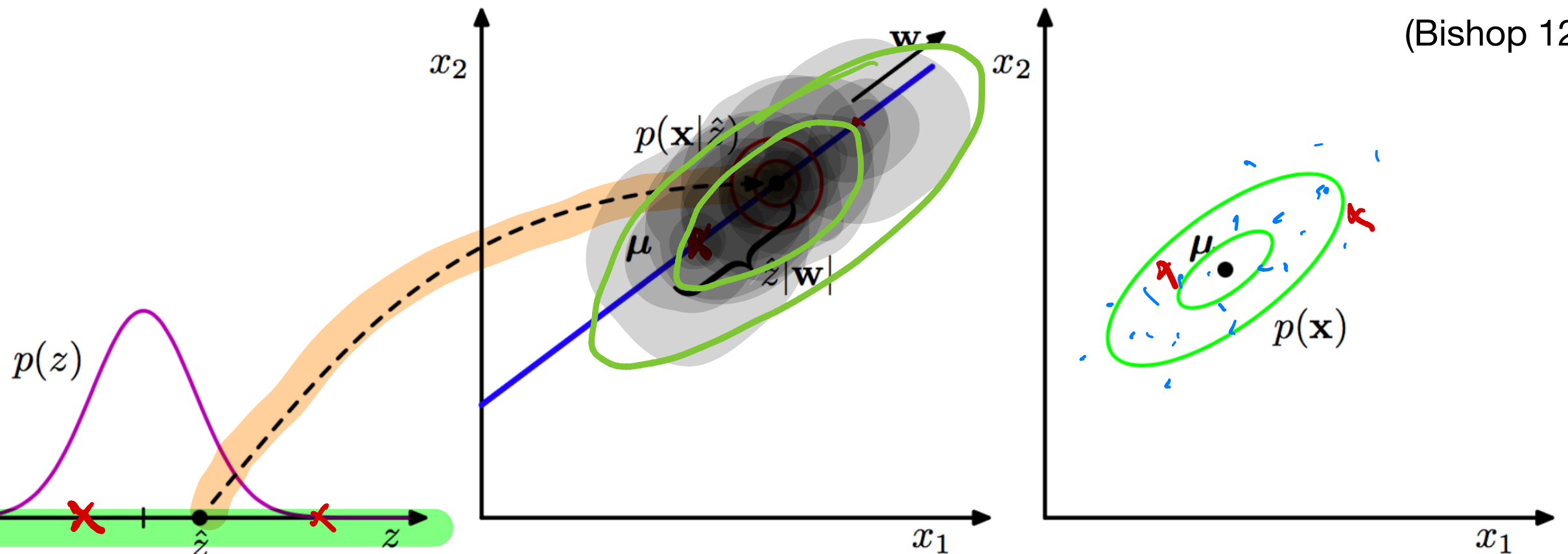
PPCA modeling assumptions

‣ The **generative model** works as follows

$$\mathbf{x} = \mathbf{W}\,\mathbf{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}$$

‣ With $\boldsymbol{\mu} \in \mathbb{R}^D$ and the **continuous latent variable** $\mathbf{z} \in \mathbb{R}^M$ with Gaussian **prior**

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}\,|\,\mathbf{0}, \mathbf{I})$$

‣ Matrix $\mathbf{W} \in R^{D \times M}$ transforms the latent variables into observed variables.

‣ The independent noise is also Gaussian

$$p(\boldsymbol{\epsilon}) = \mathcal{N}(\boldsymbol{\epsilon}\,|\,\mathbf{0}, \sigma^2 \mathbf{I})$$
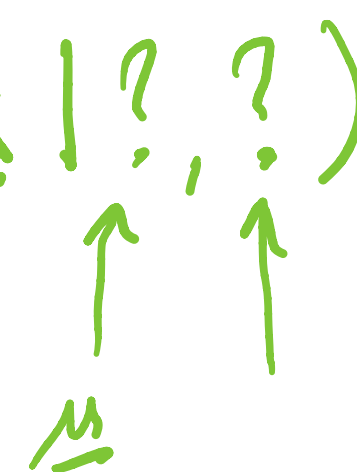
It follows...

‣ The conditional distribution of the observed variable *Gaussian!*

$$p(\mathbf{x} \mid \mathbf{z}) = \mathcal{N}(\mathbf{x} \mid \mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2 \mathbf{I})$$

‣ The marginal $p(\mathbf{x})$ is also a Gaussian (Bishop Ch. 2.3)

$$p(\mathbf{x}) = \int p(\mathbf{x}, \mathbf{z}) \mathrm{d}\mathbf{z} = \int p(\mathbf{x} \mid \mathbf{z}) p(\mathbf{z}) \mathrm{d}\mathbf{z} \; = \mathcal{N}(\mathbf{x} \mid ?, ?)$$

‣ The expected value for $\mathbf{x} \sim p(\mathbf{x})$ is given by

$$\mathbb{E}[\mathbf{x}] = \mathbb{E}[\mathbf{W}\mathbf{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}]$$
$$= \mathbf{W}\,\mathbb{E}[\mathbf{z}] + \boldsymbol{\mu} + \mathbb{E}[\boldsymbol{\epsilon}]$$
$$= \boldsymbol{\mu}$$

It follows...

▸ The covariance of $\mathbf{x} \sim p(\mathbf{x})$ is given by

$$\mathrm{Cov}[\mathbf{x}] \overset{\text{def}}{=} \mathbb{E}[(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^T]$$

$$\underset{\mathbf{x} = \mathbf{W}\mathbf{z} + \mu + \epsilon}{=} \mathbb{E}[(\mathbf{W}\mathbf{z} + \boldsymbol{\epsilon})(\mathbf{W}\mathbf{z} + \boldsymbol{\epsilon})^T]$$

$$\underset{\text{expand}}{=} \mathbb{E}[\mathbf{W}\mathbf{z}\mathbf{z}^T\mathbf{W}^T + 2\mathbf{W}\mathbf{z}\boldsymbol{\epsilon}^T + \boldsymbol{\epsilon}\boldsymbol{\epsilon}^T]$$

$$\underset{\text{linearity}}{=} \mathbf{W}\,\mathbb{E}[\mathbf{z}\mathbf{z}^T]\,\mathbf{W}^T + 2\mathbf{W}\,\mathbb{E}[\mathbf{z}\boldsymbol{\epsilon}^T] + \mathbb{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T]$$

$$= \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I} = \mathbf{C}$$

▸ Therefore

$$\boxed{p(\mathbf{x}) = \mathcal{N}(\mathbf{x} \,|\, \boldsymbol{\mu}, \mathbf{C})}$$

(handwritten annotations:)

z, ϵ independent so

$\mathrm{Cov}[z, \epsilon] = \mathbb{E}[z\epsilon^T] - \mathbb{E}[z]\mathbb{E}(\epsilon)^T = 0$

$\overset{=}{0} \qquad \overset{=}{0}$

$\Rightarrow \mathbb{E}[z\epsilon^T] = 0$

$\mathrm{Cov}[\epsilon] = \mathbb{E}[\epsilon\epsilon^T]$

$+ \mathbb{E}[\epsilon]\mathbb{E}[\epsilon]^T$

$= \sigma^2 I$

(under equation: $\underset{I}{}$, $\underset{0}{}$, $\sigma^2 I$)

Probabilistic PCA in a picture
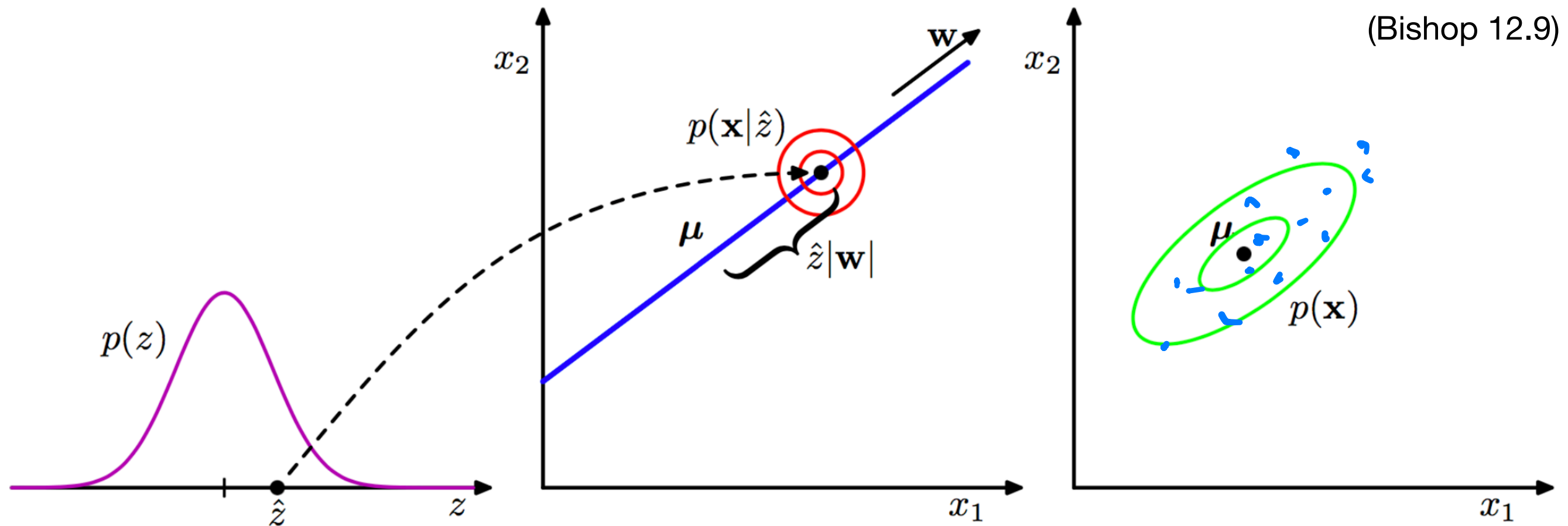
- ‣ Prior $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}\,|\,\mathbf{0}, \mathbf{I})$

- ‣ Likelihood $p(\mathbf{x}\,|\,\mathbf{z}) = \mathcal{N}(\mathbf{x}\,|\,\mathbf{W}\,\mathbf{z} + \boldsymbol{\mu}, \sigma^2 \mathbf{I})$

- ‣ Marginal $p(\mathbf{x}) = \mathcal{N}(\mathbf{x}\,|\,\boldsymbol{\mu}, \mathbf{C})$

The log-likelihood

$$\ln p(\mathbf{X} \,|\, \boldsymbol{\mu}, \mathbf{W}, \sigma^2) = \sum_{n=1}^{N} \ln p(\mathbf{x}_n \,|\, \boldsymbol{\mu}, \mathbf{W}, \sigma^2) = \sum_{n=1}^{N} \ln \mathcal{N}(\mathbf{x}_n \,|\, \boldsymbol{\mu}, \mathbf{C})$$

$$= -\frac{ND}{2} \ln(2\pi) - \frac{N}{2} \ln |\mathbf{C}| - \frac{1}{2} \sum_{n=1}^{N} (\mathbf{x}_n - \boldsymbol{\mu})^T \mathbf{C}^{-1} (\mathbf{x}_n - \boldsymbol{\mu})$$

‣ Once again, take it's derivative w.r.t. the parameter of interest, set it to zero, and solve it. For the mean:

$$\sum_{n=1}^{N} \mathbf{C}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}) = 0 \quad \Longrightarrow \quad \boldsymbol{\mu} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{x}_n$$
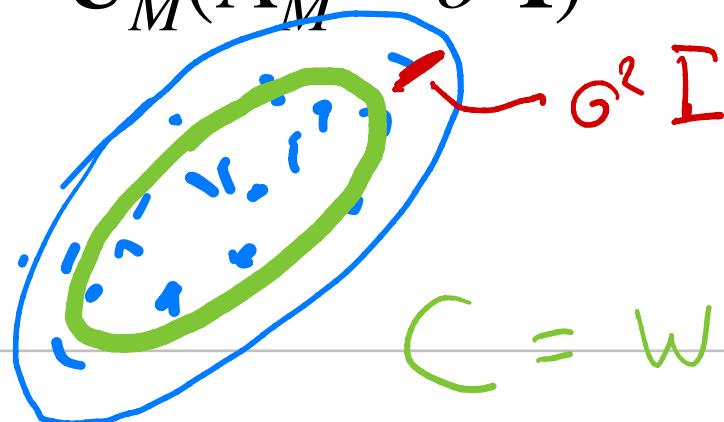
PPCA has closed-form solutions

‣ Let $\mathbf{S}$ be the sample covariance, as defined in PCA, and let its eigenvectors and eigenvalues be given

‣ The optimal parameters for the maximal log-likelihood are

$$\boldsymbol{\mu} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n$$

$$\sigma^2 = \frac{1}{D-M}\sum_{j=M+1}^{D}\lambda_j$$

average of discarded variance

$$\mathbf{W} = \mathbf{U}_M(\Lambda_M - \sigma^2\mathbf{I})^{1/2}$$

$\sim \sigma^2 I$

$C = WW^T + \sigma^2 I$

Assume $S = S_{true} + S_{noise}$

$U \wedge U^T = U_M \wedge_M U_M^T + U_- \wedge_- U_-^T$

Covariance of Gaussian was

$C = WW^T + \sigma^2 I$

C should match S_{true}

$U_M \wedge_M U_M^T = WW^T + \sigma^2 I$

$WW^T = U_M(\wedge_M + \sigma^2 I)U_M^T$

PPCA

‣ PPCA is the probabilistic **generative** version of PCA: we can also draw samples from it

‣ PPCA is a form of Gaussian distribution with number of parameters **restricted** (by the latent space)

‣ PPCA is the basis of **Bayesian PCA** (Bishop 12.2.3) in which the dimension of the latent space can be found from the data

‣ Like with the Gaussian Mixture Model, also the PPCA can be done via an **EM algorithm** (Bishop 12.2.2) - *Though not necessary because we have close form solutions*

PCA: Summary

- **Three views**

    - Max variance, min reconstruction error, probabilistic

- **Applications**

    - Dimensionality reduction

    - 2D/3D visualization

    - Compression

    - Whitening (de-correlating features)

    - (not mentioned) De-noising: discard the smallest variance features = the noise components (hopefully!)

- **Limitation**

    - Only linear transformations